SUMMER INTERNSHIP 2022 REPORT

01/06/2022 - 15/07/2022

IIT PALAKKAD

Case Study on Machine Learning Algorithms for Satellite Image Classification

Aashish R. Raheja

IIT Palakkad Summer Internship 2022

under

Dr. Deepak Jaiswal
(HOC & Assistant Professor)

Environmental Sciences and Sustainable Engineering Centre

- **INTRODUCTION** :

Technological innovations are the driving force behind advancements of all industries across the world. Amongst the technologies Data Science is playing a major role for advancement of industries . According to [1] data science is defined as an interdisciplinary field that uses scientific methods, processes, algorithms and systems to extract knowledge and insights from noisy, structured and unstructured data, and apply knowledge from data across a broad range of application domains. Data science is related to data mining, machine learning and big data. Machine Learning, a part of Data Science, has various applications such as Sentiment analysis, Image Classification, Prediction of trends in different industries, Speech Recognition etc. Machine Learning is mainly divided into 2 categories: Supervised Learning and Unsupervised Learning.

According to [2] Supervised Learning is defined as the type of machine learning in which machines are trained using well "labelled" training data, and on the basis of that data, machines predict the output. The labelled data means some input data is already tagged with the correct output. Examples of Supervised Learning algorithms are, Decision Tree, Random Forest etc. According to [3], Unsupervised Learning is defined as , unsupervised learning is a machine learning technique in which models are not supervised using a training dataset. Instead, models itself find the hidden patterns and insights from the given data. Examples of Unsupervised Learning algorithms are K-Means clustering, Neural Networks etc. One of the major applications of Machine Learning is image classification. Image classification is a type of classification in which components or the whole image can be uniquely identified by a machine. Image classification has many applications in fields such as medical, remote sensing, traffic analysis, autonomous driving etc.

One of the major applications of image classification is in remote sensing. According to [4], Satellite images have many applications in meteorology, oceanography, fishing, agriculture, biodiversity conservation, forestry, landscape, geology, cartography, regional planning, education, intelligence and warfare. An important aspect of satellite image classification is to identify land use by using land cover classification. Various Machine Learning and Deep Learning techniques have been used for classification of Land cover. Land use classification is done by creating a dataset of a geographical location by dividing it into a number of classes and then classifying a new area using the pixel values for different types of bands obtained in the original dataset. Existing work in this domain is done by various methods such as [5] have done classification of satellite imagery by developing CNN models and thus segmented image of the original data is produced according to classes. In [6] a comparative study is done for two methods for image classification namely Maximum Likelihood for supervised classification and ISODATA clustering for unsupervised classification.

Our work aims to do a comparative study to evaluate the accuracy of Machine Learning models for classification of satellite image data. The dataset for testing and training is prepared from scratch and after preprocessing the models are trained to evaluate performance. Algorithms implemented include Random Forest, Support Vector Classifier, Decision Tree, Gaussian Naive Bayes Classifier ,XgBoost Classifier and Stochastic Gradient Descent.

STUDY AREA :

The study area is selected from the state of Kerala in India and has coordinate boundaries as follows:

Upper left - 76.537812 10.861687 Decimal Degrees.

Upper right - 76.780173 10.861447 Decimal Degrees.

Lower left - 76.537812 10.692130 Decimal Degrees.

Lower right - 76.780412 10.692370 Decimal Degrees.

- **DATA USED :**

In order to perform image classification of satellite images, a sample raster of a geographical location is obtained using the USGS website [7] . This raster consists of LANDSAT-8 data. The data to be classified consists of 6 bands and has a GeoTIF file format.

The data on which the model is trained and builts is created using Google Earth . This data is in .shp format or also called a shapefile. It is a polygon shapefile which consists of 103 polygons and each polygon has a class assigned to each for classification. The classes present are : Vegetation, Agriculture Land, Water Body, Settlement ,Rocky Region , Barren Land.

The base raster for classification is (Fig-1) :

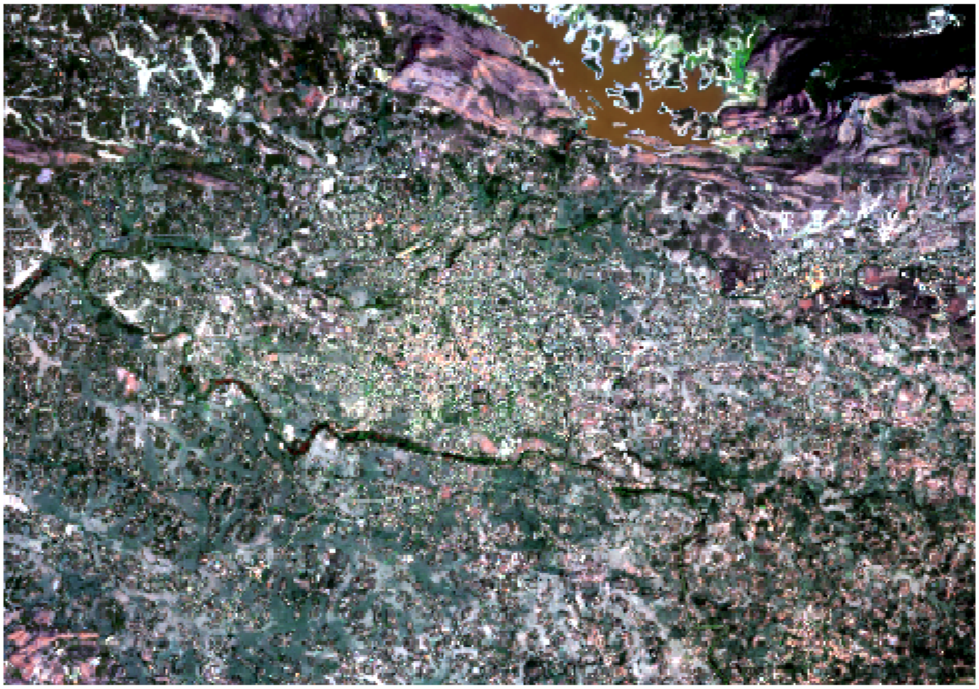
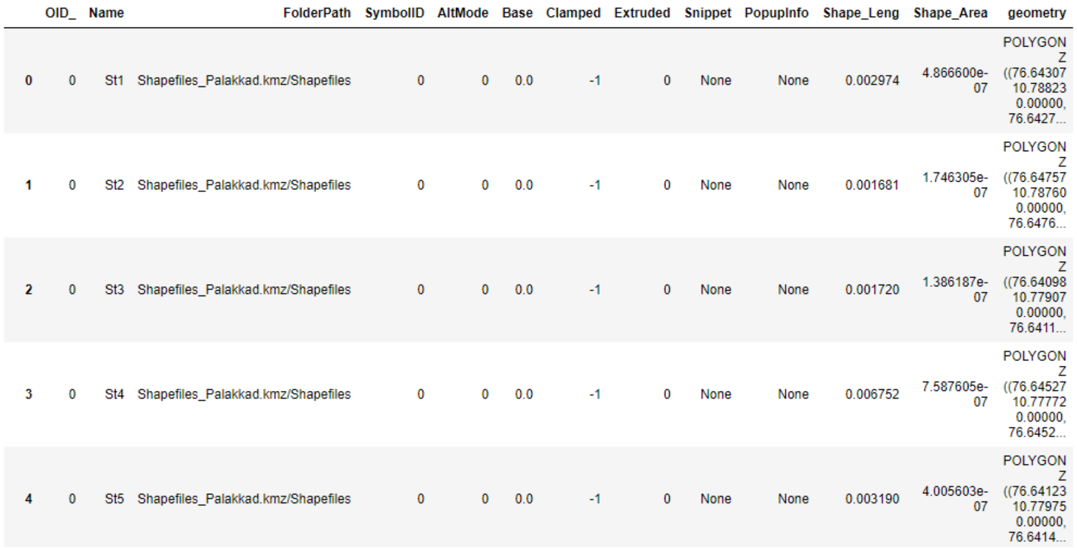


Fig-1

The Shapefile used for training is : (Fig-2)



	OID_	Name	FolderPath	SymbolID	AltMode	Base	Clamped	Extruded	Snippet	PopupInfo	Shape_Leng	Shape_Area	geometry
0	0	St1	Shapefiles_Palakkad.kmz/Shapefiles	0	0	0.0	-1	0	None	None	0.002974	4.866600e-07	POLYGON Z ((76.64307 10.78823 0.00000, 76.6427...
1	0	St2	Shapefiles_Palakkad.kmz/Shapefiles	0	0	0.0	-1	0	None	None	0.001681	1.746305e-07	POLYGON Z ((76.64757 10.78760 0.00000, 76.6476...
2	0	St3	Shapefiles_Palakkad.kmz/Shapefiles	0	0	0.0	-1	0	None	None	0.001720	1.386187e-07	POLYGON Z ((76.64098 10.77907 0.00000, 76.6411...
3	0	St4	Shapefiles_Palakkad.kmz/Shapefiles	0	0	0.0	-1	0	None	None	0.006752	7.587605e-07	POLYGON Z ((76.64527 10.77772 0.00000, 76.6452...
4	0	St5	Shapefiles_Palakkad.kmz/Shapefiles	0	0	0.0	-1	0	None	None	0.003190	4.005603e-07	POLYGON Z ((76.64123 10.77975 0.00000, 76.6414...

Fig-2

Shapefile Representation : (Fig-3)

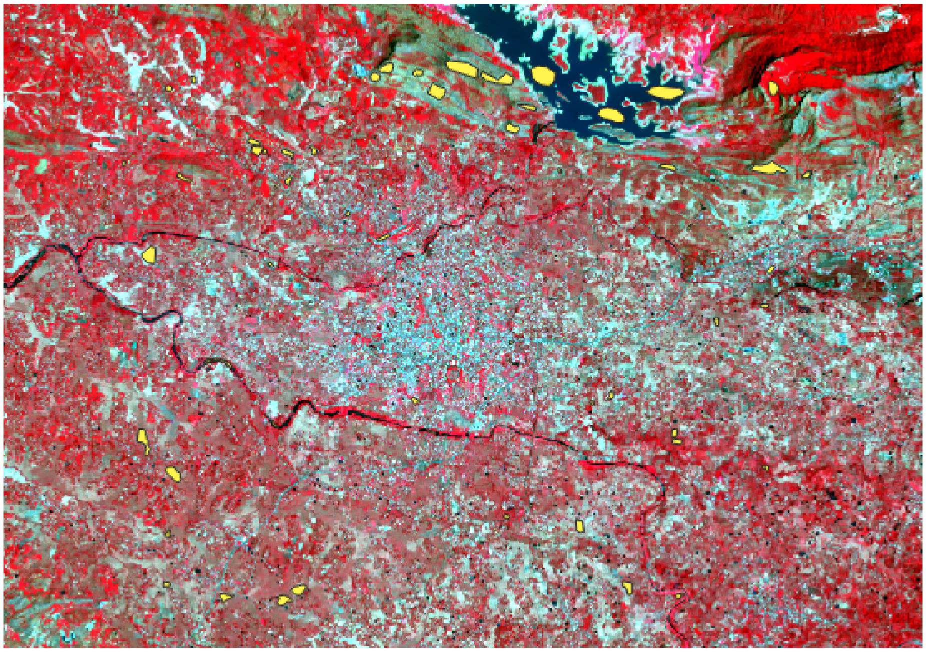


Fig-3

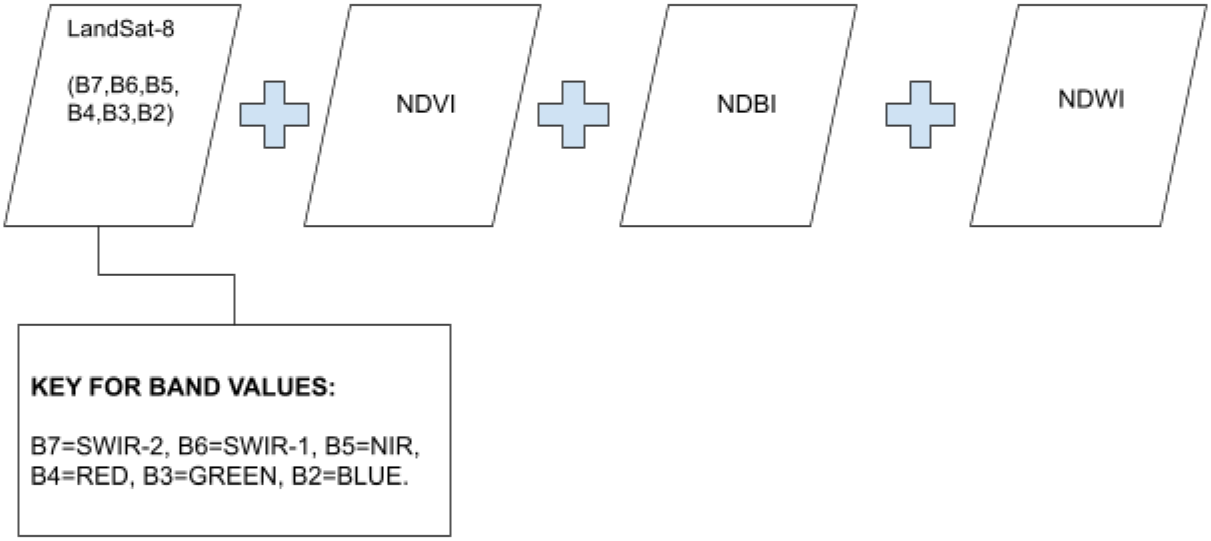
Polygons can be seen in Yellow colour.

- **METHODOLOGY :**

1. DATA PREPROCESSING :

   The first part of the project focuses on data pre-processing and handling of raw data so that it can be fed into the model.

   In order to improve the pixel strength of some of the classes in our base raster, we calculated three indexes namely Normalised Difference Built-up Index (NDBI) , Normalised Difference Water Index (NDWI), and Normalised Difference Vegetation Index (NDVI). Using Python and GDAL a new raster was formed by stacking the array values of the Original raster ,NDBI,NDWI and NDVI (Fig-4).



KEY FOR BAND VALUES:

B7=SWIR-2, B6=SWIR-1, B5=NIR, B4=RED, B3=GREEN, B2=BLUE.

Fig-4

Now, all the processing is done on the original raster. Consider 6 bands+NDVI NDWI+NDBI stacks. GDAL was used to read Raster data to get its attributes like height, width, Number of bands etc,. Each individual band was read as an array using the GDAL library and the shape of array was 3D in the form of
(Number of bands, Rows, Columns) (Fig-5).

In Order to feed data into the model we needed a 2D array, so conversion of the shape of the array for each band was done. The new shape is ((rows*columns),NumberBands).

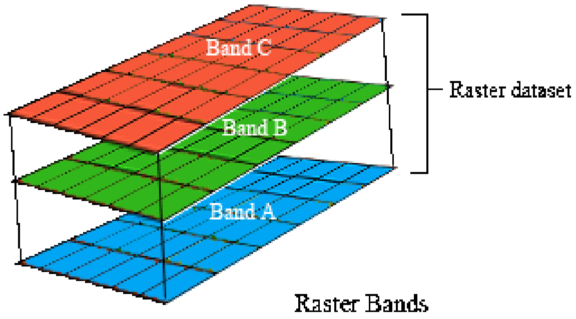


Fig-5

The original raster contains certain aspects such as its Coordinate Reference System, Projection information , Bounds of the raster etc which is accessed using the Python GDAL Library.

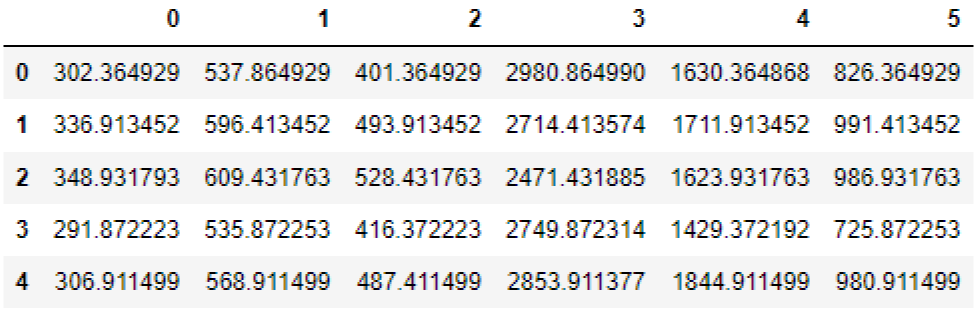
2. MODEL TRAINING:

In order to train our models, first some processing and handling of data is done.

STEPS :-

>The raster is read again using GDAL, the shape is again converted from (Number of bands, Rows, Columns) to ((rows*columns),Number of Bands).

> Then a pandas dataframe is created from the array as shown in Fig “ 6“

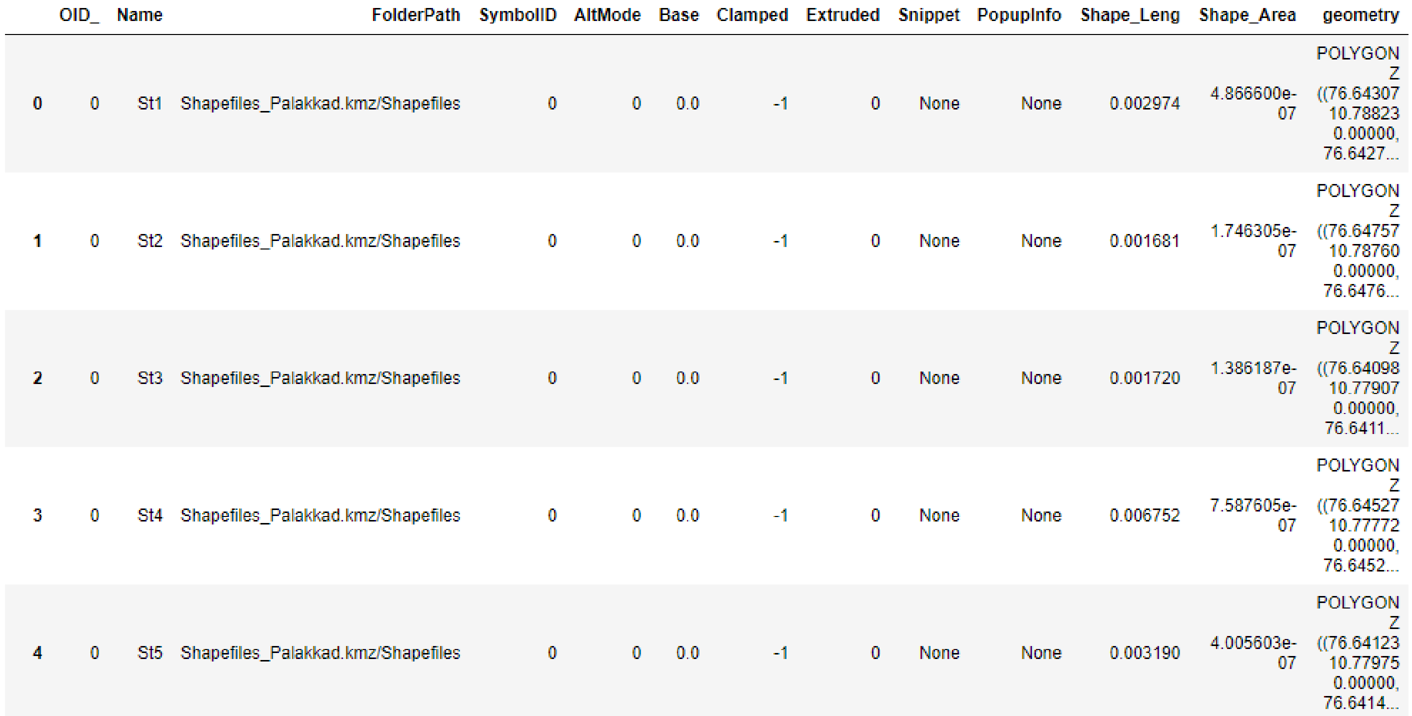


	0	1	2	3	4	5
0	302.364929	537.864929	401.364929	2980.864990	1630.364868	826.364929
1	336.913452	596.413452	493.913452	2714.413574	1711.913452	991.413452
2	348.931793	609.431763	528.431763	2471.431885	1623.931763	986.931763
3	291.872223	535.872253	416.372223	2749.872314	1429.372192	725.872253
4	306.911499	568.911499	487.411499	2853.911377	1844.911499	980.911499

Fig “ 6“ - Pandas Dataframe

>Now how shape file is handled:

a. The original shapefile is received when it is read using geopandas as shown in Fig-7.



	OID_	Name	FolderPath	SymbolID	AltMode	Base	Clamped	Extruded	Snippet	PopupInfo	Shape_Leng	Shape_Area	geometry
0	0	St1	Shapefiles_Palakkad.kmz/Shapefiles	0	0	0.0	-1	0	None	None	0.002974	4.866600e-07	POLYGON Z ((76.64307 10.78823 0.00000, 76.6427...
1	0	St2	Shapefiles_Palakkad.kmz/Shapefiles	0	0	0.0	-1	0	None	None	0.001681	1.746305e-07	POLYGON Z ((76.64757 10.78760 0.00000, 76.6476...
2	0	St3	Shapefiles_Palakkad.kmz/Shapefiles	0	0	0.0	-1	0	None	None	0.001720	1.386187e-07	POLYGON Z ((76.64098 10.77907 0.00000, 76.6411...
3	0	St4	Shapefiles_Palakkad.kmz/Shapefiles	0	0	0.0	-1	0	None	None	0.006752	7.587605e-07	POLYGON Z ((76.64527 10.77772 0.00000, 76.6452...
4	0	St5	Shapefiles_Palakkad.kmz/Shapefiles	0	0	0.0	-1	0	None	None	0.003190	4.005603e-07	POLYGON Z ((76.64123 10.77975 0.00000, 76.6414...

Fig-7

The above geopandas dataframe consists of 103 rows, i.e it contains 103 polygons.

b. In 103 polygons, each polygon consists of varying numbers of coordinates, when acsessed using a loop add up to 1674 points.

c. Then the geometry of the shapefile is changed to point by extraction of coordinates using geopandas and mapping them to the same index.

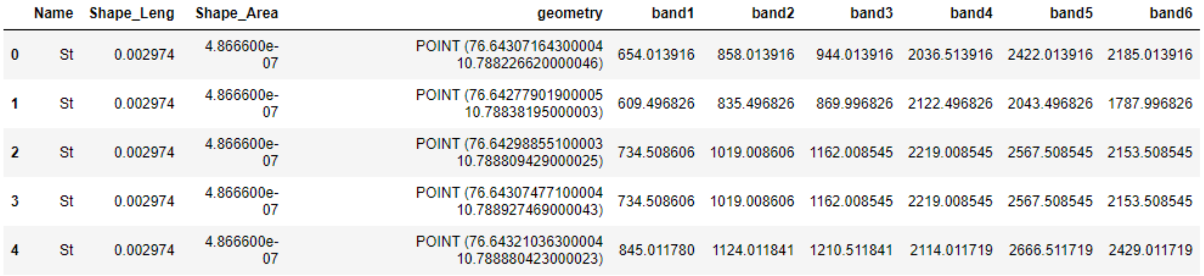
d. Using the rasterio library the original raster values are read and then extraction of band values is done for every coordinate and a separate data frame for band values is created.

e. Then Point data frame and band dataframe are concatenated together to give our original dataset.

f. Since data categories for general classes are named with numbers like St1,St2, etc, these all are then converted into general categories like St,Vg etc.

g. Model Training:

- Now, our training dataset looks like (Fig-8):



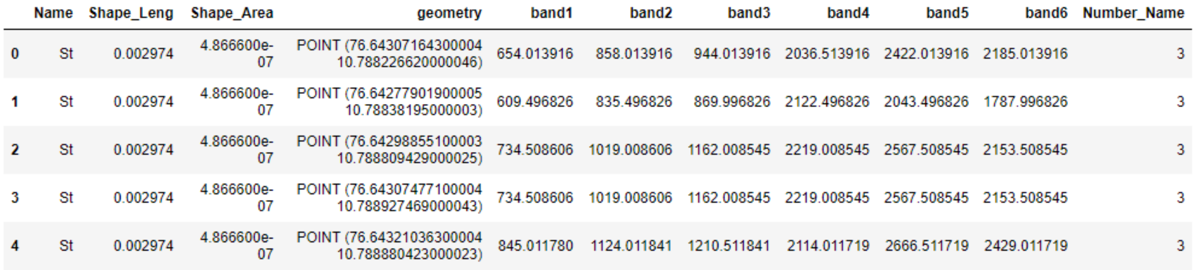
	Name	Shape_Leng	Shape_Area	geometry	band1	band2	band3	band4	band5	band6
0	St	0.002974	4.866600e-07	POINT (76.64307164300004 10.788226620000046)	654.013916	858.013916	944.013916	2036.513916	2422.013916	2185.013916
1	St	0.002974	4.866600e-07	POINT (76.64277901900005 10.78838195000003)	609.496826	835.496826	869.996826	2122.496826	2043.496826	1787.996826
2	St	0.002974	4.866600e-07	POINT (76.64298855100003 10.788809429000025)	734.508606	1019.008606	1162.008545	2219.008545	2567.508545	2153.508545
3	St	0.002974	4.866600e-07	POINT (76.64307477100004 10.788927469000043)	734.508606	1019.008606	1162.008545	2219.008545	2567.508545	2153.508545
4	St	0.002974	4.866600e-07	POINT (76.64321036300004 10.788880423000023)	845.011780	1124.011841	1210.511841	2114.011719	2666.511719	2429.011719

Fig-8

- Our dependent variable is Name and independent variables are band1,band2,band3,band4,band5 and band6.

- Since our Name column is of type string and model cannot classify string values, we need to encode them using a library from sklearn known as label encoder.

- After encoding (Fig-9):



	Name	Shape_Leng	Shape_Area	geometry	band1	band2	band3	band4	band5	band6	Number_Name
0	St	0.002974	4.866600e-07	POINT (76.64307164300004 10.788226620000046)	654.013916	858.013916	944.013916	2036.513916	2422.013916	2185.013916	3
1	St	0.002974	4.866600e-07	POINT (76.64277901900005 10.78838195000003)	609.496826	835.496826	869.996826	2122.496826	2043.496826	1787.996826	3
2	St	0.002974	4.866600e-07	POINT (76.64298855100003 10.788809429000025)	734.508606	1019.008606	1162.008545	2219.008545	2567.508545	2153.508545	3
3	St	0.002974	4.866600e-07	POINT (76.64307477100004 10.788927469000043)	734.508606	1019.008606	1162.008545	2219.008545	2567.508545	2153.508545	3
4	St	0.002974	4.866600e-07	POINT (76.64321036300004 10.788880423000023)	845.011780	1124.011841	1210.511841	2114.011719	2666.511719	2429.011719	3

Fig-9

- Now for fitting or training the model, we use the Sklearn library to split the dataset in a train:test ratio of 80:20.

- Therefore from the shapefile dataframe, we select the dependent and independent variable columns and train our model respectively and then classify our original raster using its band values dataframe.

- After fitting of the model and then classifying the original raster, we get the classes in the form of an array, we change the shape of the array into (rows*columns) because we need to plot it and write it again on a new GeoTIFF file.

- This operation is carried out using the GDAL library, and CRS, Projections and other attributes are obtained from the original raster.

> A graph is drawn from data collected from [8], which shows which algorithms are used for satellite image classification (Fig-10).

Keywords used for searching were : Satellite Image Classification and Algorithm Name :-

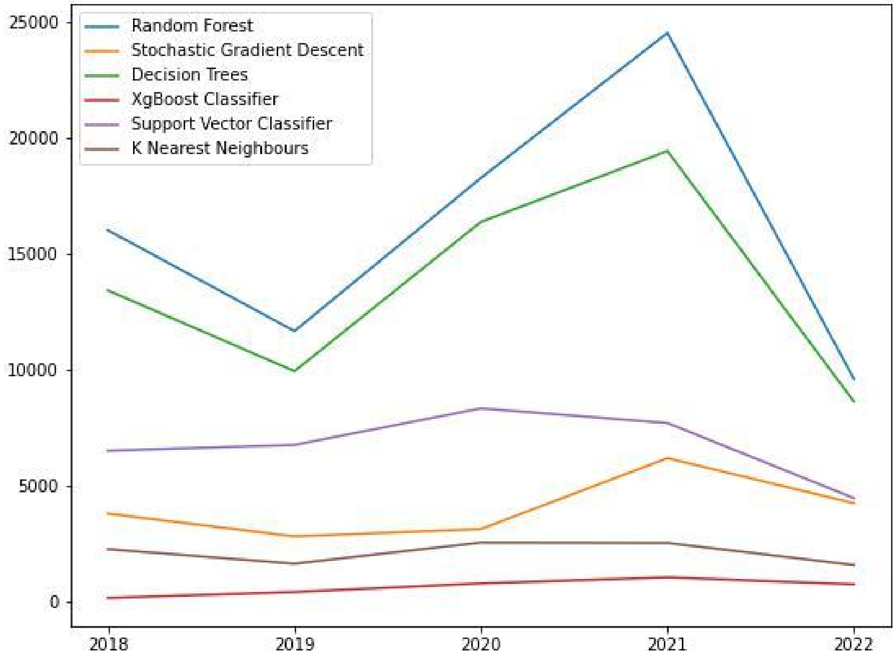


Fig - 10

- **RESULTS :**

Here is a sample classified result from XgBoost Classifier (Fig-11):

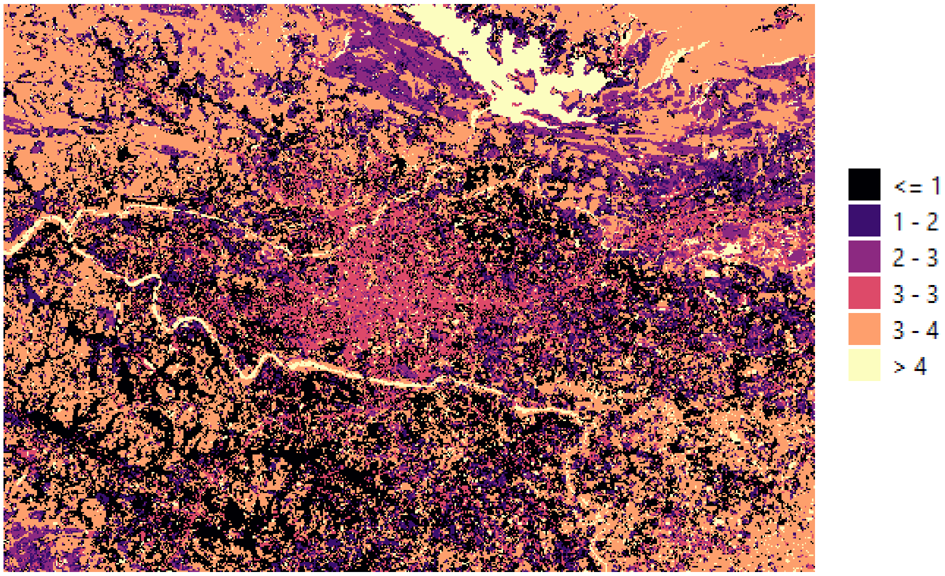


Fig-11

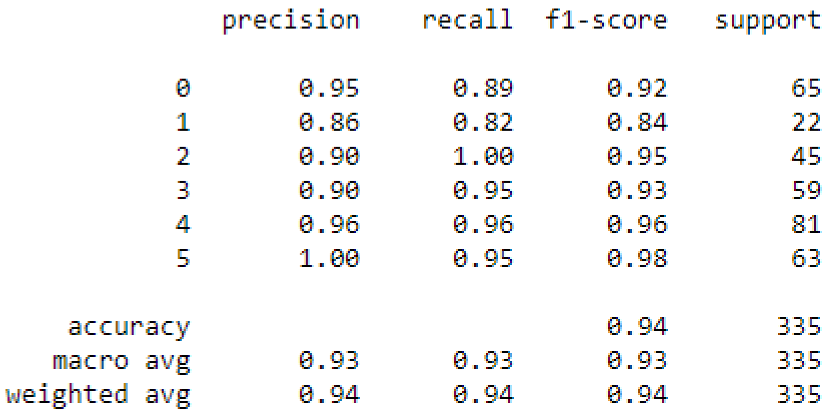
Agriculture : <=1 , Barren : 1-2 , Rocky : 2-3 , Settlement : 3 , Vegetation : 3-4 , Wetland : >4

6 models were trained and results are as follows:

- XgBoost Classifier:

  XgBoost Classifier has an accuracy of 0.94.

  Confusion Matrix and Accuracy Report (Fig-12):



```
              precision    recall  f1-score   support

           0       0.95      0.89      0.92        65
           1       0.86      0.82      0.84        22
           2       0.90      1.00      0.95        45
           3       0.90      0.95      0.93        59
           4       0.96      0.96      0.96        81
           5       1.00      0.95      0.98        63

    accuracy                           0.94       335
   macro avg       0.93      0.93      0.93       335
weighted avg       0.94      0.94      0.94       335
```

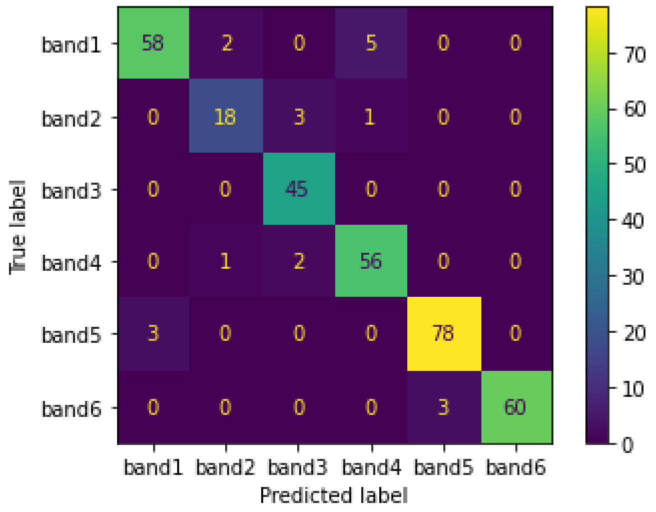


Fig-12

- Random Forest:

  Random Forest has an accuracy of 0.91 (Fig-14).

  Here is a plot of tuning the parameter n_estimator of random forest to get maximum accuracy (Fig-13).

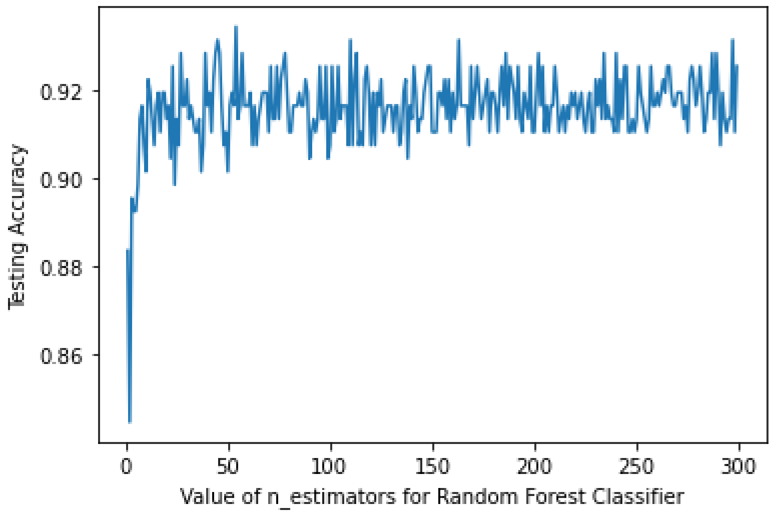
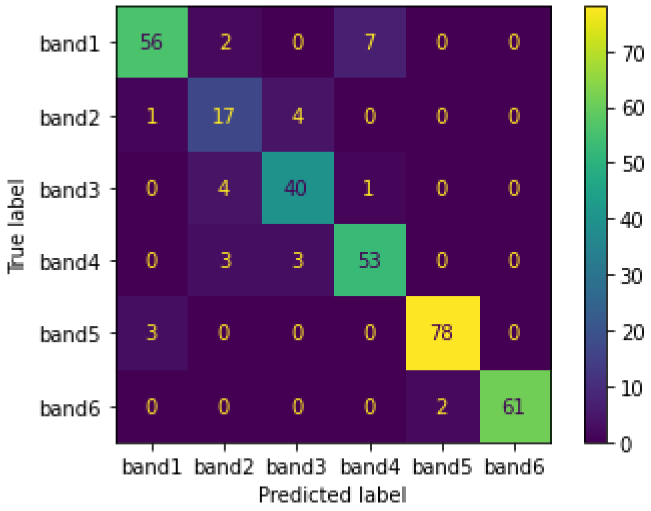
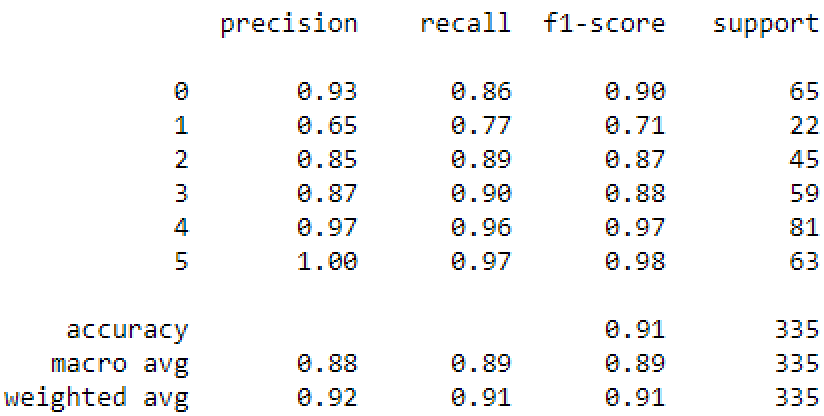


Fig-13

- Confusion Matrix and Accuracy Report:



True label \ Predicted label	band1	band2	band3	band4	band5	band6
band1	56	2	0	7	0	0
band2	1	17	4	0	0	0
band3	0	4	40	1	0	0
band4	0	3	3	53	0	0
band5	3	0	0	0	78	0
band6	0	0	0	0	2	61



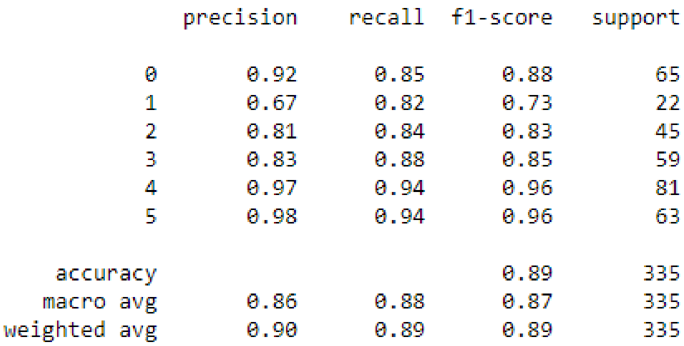
```
              precision    recall  f1-score   support

           0       0.93      0.86      0.90        65
           1       0.65      0.77      0.71        22
           2       0.85      0.89      0.87        45
           3       0.87      0.90      0.88        59
           4       0.97      0.96      0.97        81
           5       1.00      0.97      0.98        63

    accuracy                           0.91       335
   macro avg       0.88      0.89      0.89       335
weighted avg       0.92      0.91      0.91       335
```

Fig-14

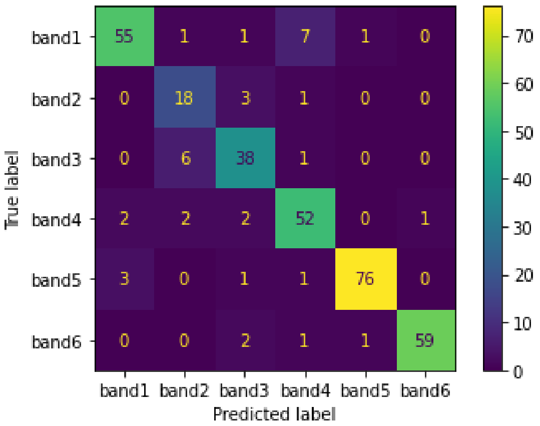
2. Decision Trees:

Decision Tree has an accuracy of 0.89 (Fig-15)

Confusion Matrix and Accuracy Report:



```
              precision    recall  f1-score   support

           0       0.92      0.85      0.88        65
           1       0.67      0.82      0.73        22
           2       0.81      0.84      0.83        45
           3       0.83      0.88      0.85        59
           4       0.97      0.94      0.96        81
           5       0.98      0.94      0.96        63

    accuracy                           0.89       335
   macro avg       0.86      0.88      0.87       335
weighted avg       0.90      0.89      0.89       335
```



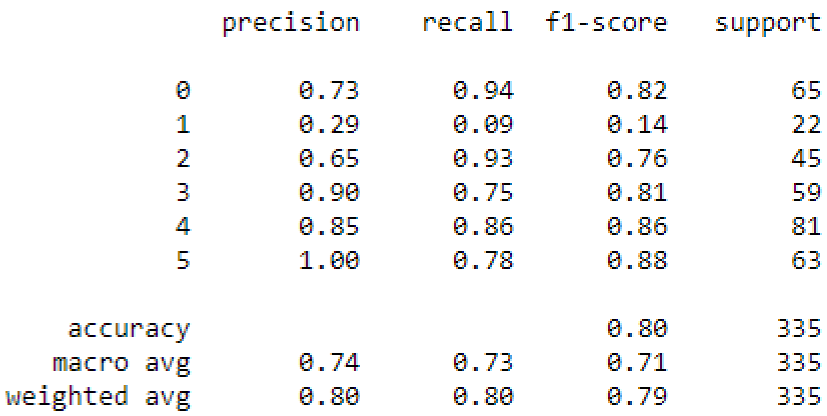
	band1	band2	band3	band4	band5	band6
band1	55	1	1	7	1	0
band2	0	18	3	1	0	0
band3	0	6	38	1	0	0
band4	2	2	2	52	0	1
band5	3	0	1	1	76	0
band6	0	0	2	1	1	59

Fig-15

- Support Vector Classifier:

SVC has an accuracy of 0.8

Confusion Matrix and Accuracy Report for SVC: (Fig-16)



```
              precision    recall  f1-score   support

           0       0.73      0.94      0.82        65
           1       0.29      0.09      0.14        22
           2       0.65      0.93      0.76        45
           3       0.90      0.75      0.81        59
           4       0.85      0.86      0.86        81
           5       1.00      0.78      0.88        63

    accuracy                           0.80       335
   macro avg       0.74      0.73      0.71       335
weighted avg       0.80      0.80      0.79       335
```

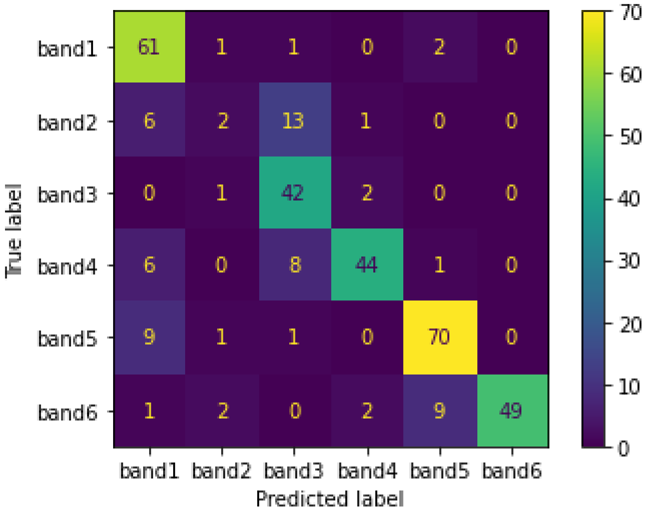
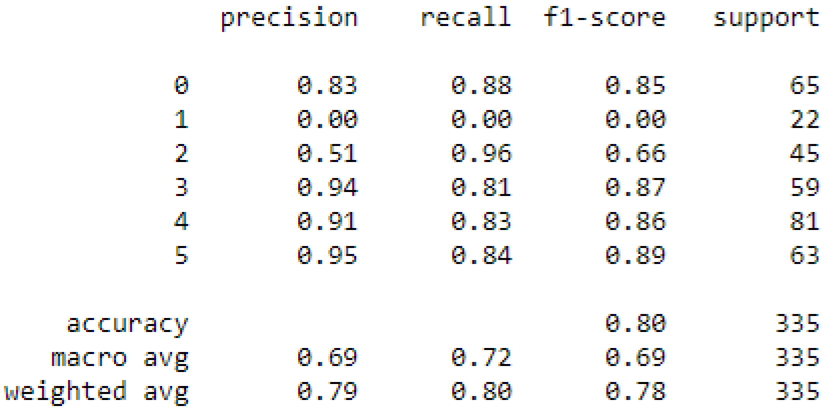


Fig-16

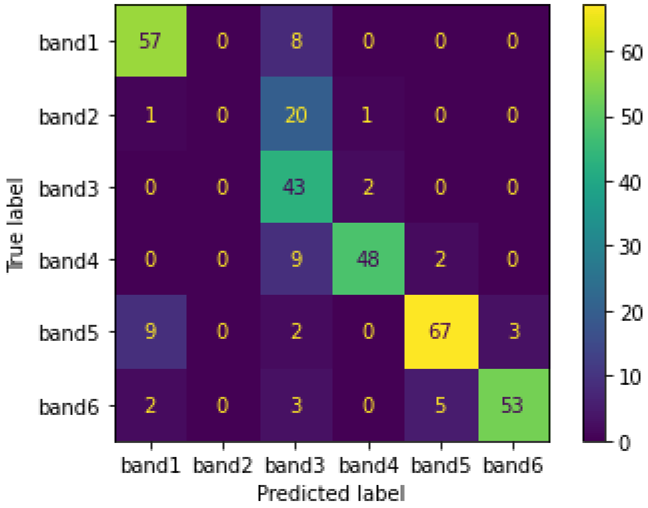
- Stochastic Gradient Descent:

SGD has an accuracy of 0.8

Confusion Matrix and Accuracy Report for SGD: (Fig-17)



```
              precision    recall  f1-score   support

           0       0.83      0.88      0.85        65
           1       0.00      0.00      0.00        22
           2       0.51      0.96      0.66        45
           3       0.94      0.81      0.87        59
           4       0.91      0.83      0.86        81
           5       0.95      0.84      0.89        63

    accuracy                           0.80       335
   macro avg       0.69      0.72      0.69       335
weighted avg       0.79      0.80      0.78       335
```



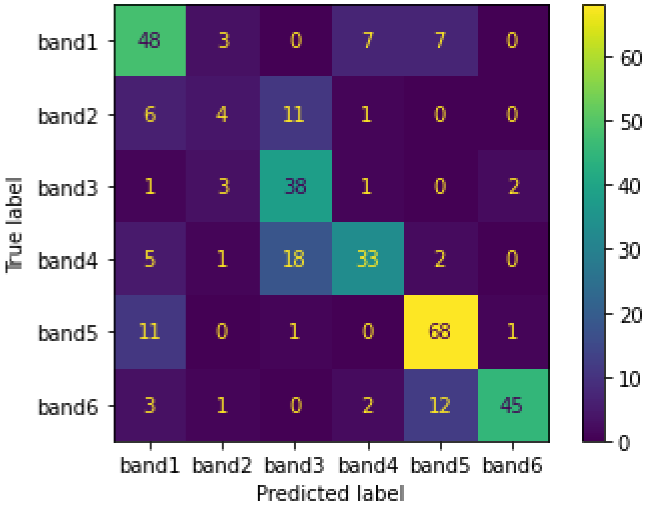
True label \ Predicted label	band1	band2	band3	band4	band5	band6
band1	57	0	8	0	0	0
band2	1	0	20	1	0	0
band3	0	0	43	2	0	0
band4	0	0	9	48	2	0
band5	9	0	2	0	67	3
band6	2	0	3	0	5	53

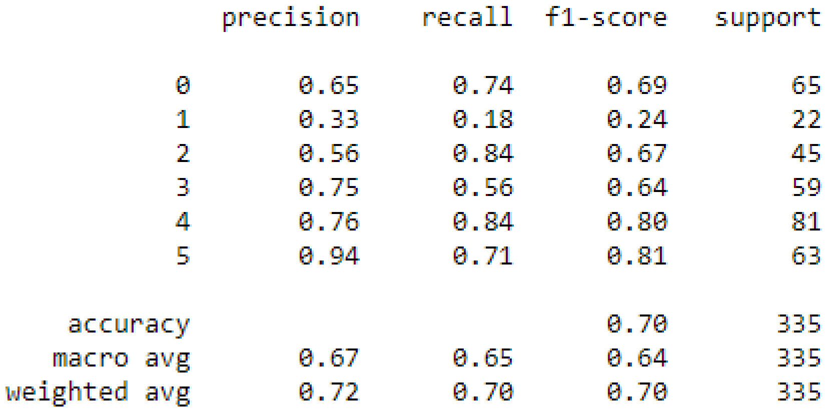
Fig-17

- Gaussian Naive Bayes:

GNB has an accuracy of 0.7

Confusion Matrix and Accuracy Report for GNB: (Fig-18)





```
              precision    recall  f1-score   support

           0       0.65      0.74      0.69        65
           1       0.33      0.18      0.24        22
           2       0.56      0.84      0.67        45
           3       0.75      0.56      0.64        59
           4       0.76      0.84      0.80        81
           5       0.94      0.71      0.81        63

    accuracy                           0.70       335
   macro avg       0.67      0.65      0.64       335
weighted avg       0.72      0.70      0.70       335
```

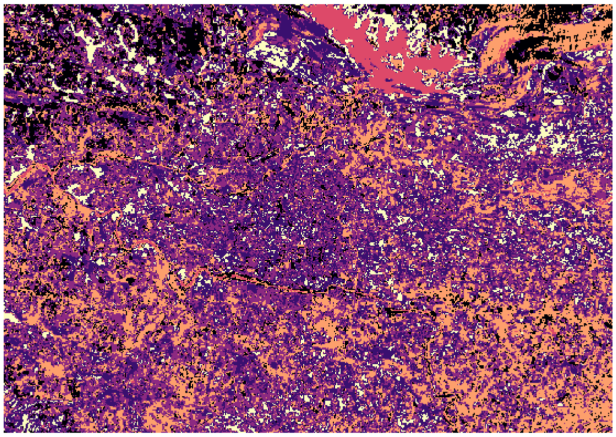
Fig-18

- **METHODS THAT DID NOT WORK :**

1. K-Means: K-Means clustering is a type of Unsupervised Machine Learning algorithm that forms clusters of data based on similar classes.

   K-Means is one of the least used algorithms for satellite image classification as represented in the graph above.

   When K-Means was fitted on the dataset, the results initially looked good but when Supervised classification was performed and the output rasters were compared visually, it was seen that the output from KMeans was not clear and contained misclassifications as compared to Supervised Learning output.



--

Fig 19 - K MEANS

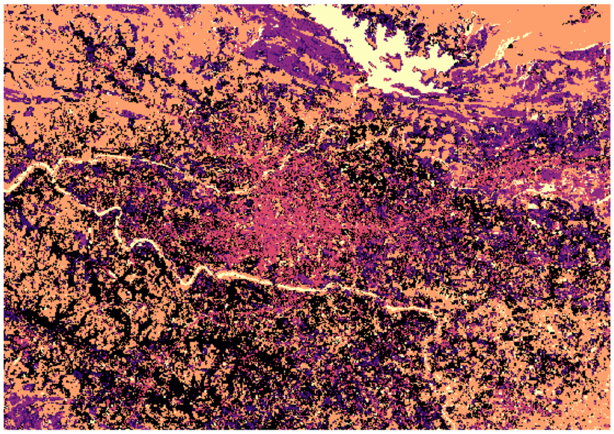


Fig-20 - XGBOOST

When you compare both the outputs, you can see the river part i.e Wetland class is clearly visible in Xgboost but it is not seen in K-Means and the river gets merged with the Settlement class.

2. Deep Learning:

   Deep Learning classification methods were also tried for classification of raster image.

   An Artificial Neural Network Model was built using Tensorflow and Keras, on the training dataset.

   As stated earlier, our training labelled dataset contains 1674 coordinates and our raster to classify contains 57328 coordinates.

   Now when the Neural Net was built it was observed that the model was overfitting i.e it was unevenly classifying our raster.

   Inorder to solve this, equal sampling of every class in our training dataset was done, that did not work, then increasing and decreasing our neural network layers, number of neurons and activation functions was also done and yet the results were the same.

The conclusion to this problem can be seen as, due to the small size of our training dataset, the model is not able to learn efficiently and instead of learning, it is memorising the values. That is why it is not able to handle large dataset classification.

OUTPUT FROM DEEP LEARNING (Fig-21):

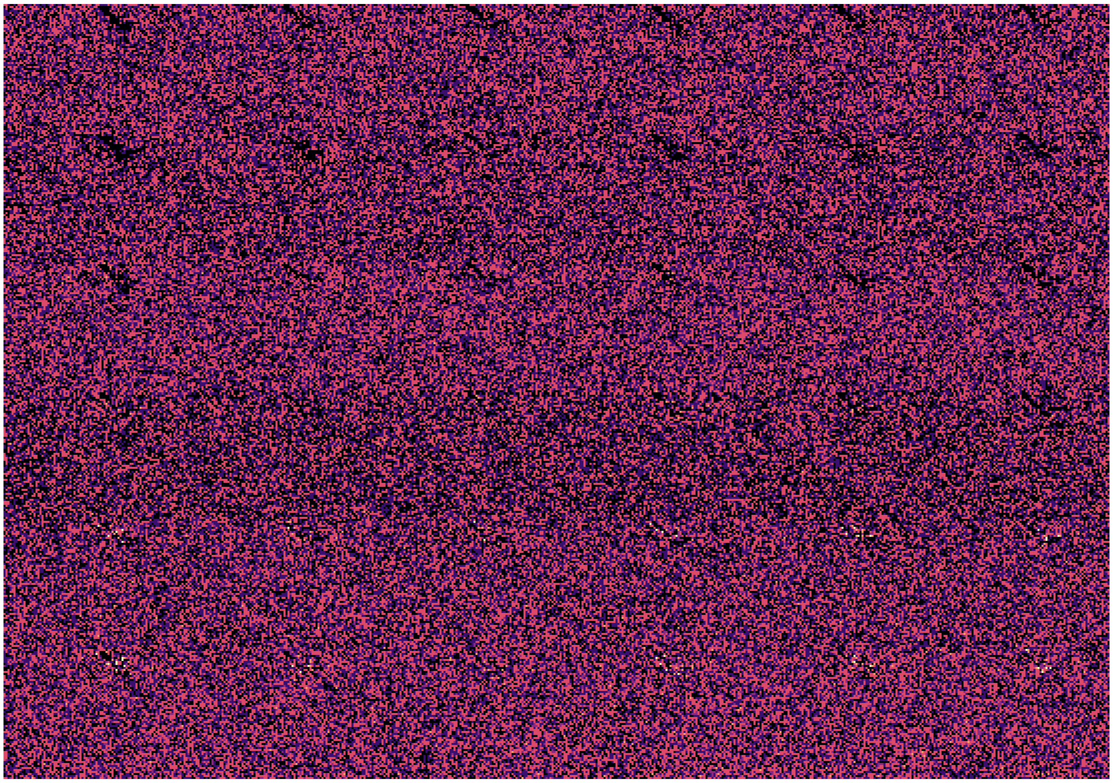


Fig-21

One of the possible solutions can be to perform accurate hyperparameter tuning and increase the size of the training dataset, also selection of correct activation function, neural layers and number of neurons in each layer could get the desired result.

- **CONCLUSION :**

Thus after classifying the data using all the models, XgBoost Classifier yielded the highest accuracy of 94% followed by Random Forest with 91% and Naive Bayes had the lowest accuracy of 70% (Fig-22).

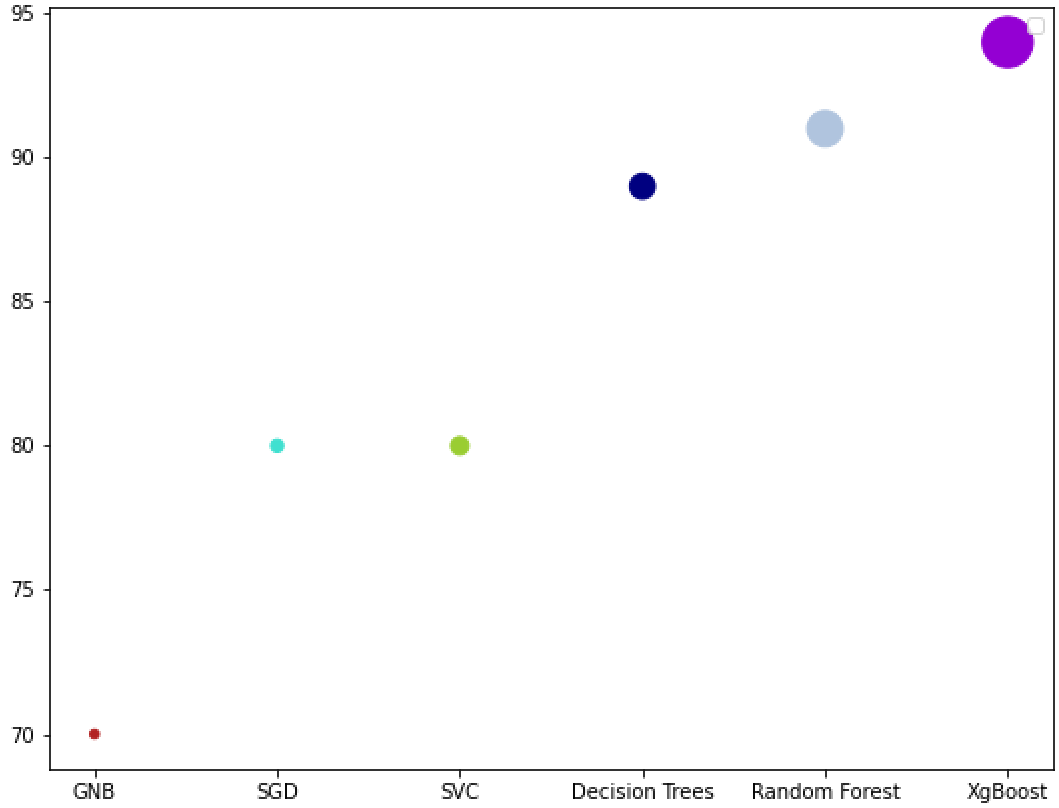


Fig-22

- NOTE:

As noticed from all the confusion matrices generated above, it can be seen that values of band-2 in all models do not correlate properly as compared to other bands.

As a result a small experiment was carried out i.e model training was done by eliminating the band-2 data from our training dataset and band-2 data was also eliminated from the original raster to classify.

The models built were Random Forest and XgBoost. The accuracy results can be seen in the accuracy table below.

Table-1

MODELS	6 BANDS	5 BANDS (without band-2)
Random Forest	0.91	0.89
XgBoost	0.94	0.90

REFERENCES :

1. https://en.wikipedia.org/wiki/Data_science

2. https://www.javatpoint.com/supervised-machine-learning

3. https://www.javatpoint.com/unsupervised-machine-learning

4. https://en.wikipedia.org/wiki/Satellite_imagery

5.Spoorthi D. M1 , Suresh Kumar M. Classification of Satellite Images .Department of Information Science, Dayananda Sagar College of Engineering, Bangalore, India, 06.2016-67962946/2021.7340

6.Ahmad, A., & Quegan, S. (2013). Comparative analysis of supervised and unsupervised classification on multispectral data. *Applied Mathematical Sciences*, 7(74), 3681-3694.

7.https://www.usgs.gov/

8.https://app.dimensions.ai/discover/publication